

PROGRAM SIMULASI PERHITUNGAN POPULASI FLUKS NEUTRON DALAM TERAS REAKTOR NUKLIR

Bagus Tri Atmoyo¹, Syarip², Supriyono¹

¹Sekolah Tinggi Teknologi Nuklir-Badan Tenaga Nuklir Nasional

²Pusat Teknologi Akselerator dan Proses Bahan-Badan tenaga Nuklir Nasional

Jl. Babarsari Kotak Pos 6101/YKBB Yogyakarta

Email : bagustriatmoyo@gmail.com

ABSTRAK

Program simulasi perhitungan fluks neutron dalam teras reaktor nuklir memiliki input berupa fraksi neutron kasip (β), konstanta peluruhan (λ), reaktivitas (ρ), umur satu generasi neutron (l) dan waktu (t). Data tersebut diolah untuk menyusun suatu persamaan karakteristik guna memperoleh akar-akar persamaan karakteristik (w_j) dan residu (A_j). w_j dan A_j digunakan dalam membentuk persamaan populasi fluks neutron terhadap fungsi waktu (n/n_0). Output dari program ini berupa populasi fluks neutron terhadap fungsi waktu yang ditampilkan dalam bentuk angka maupun grafik. Pengujian dilakukan salah satunya dengan memasukkan data β dan λ dari isotop U^{235} , ρ dan l masing-masing sebesar 0,002 dan 0,001 s. Hasil pengujian menunjukkan nilai n/n_0 saat $t = 0$ s sebesar 0,07088 dan mengalami peningkatan yang signifikan saat $t = 55$ s menjadi 1,31444 dan saat $t = 150$ s menjadi 167,21324. Hal ini sesuai dengan pernyataan bahwa bila $\rho > 0$ maka grafik populasi fluks neutron terhadap waktu cenderung menjauhi nol.

Kata Kunci : Simulasi, Reaktivitas, Persamaan Karakteristik, Fluks Neutron, Fungsi Waktu.

ABSTRACT

Simulation program for calculating the neutron flux in the nuclear reactor core have input in form of delayed neutron fraction (β), the decay constant (λ), reactivity (ρ), the age of a generation of neutron (l) and time (t). The data is processed to construct a characteristic equation to obtain its roots (w_j) and residue (A_j). w_j and A_j are used in forming the population of neutron flux equation versus a function of time (n/n_0). The output of this program is a population of neutron flux to the function of the time displayed whether graphs or numbers or both. One of the test performed by entering the β and λ from the U^{235} isotope, ρ and l shown as 0.003 and 0.0001 s respectively. The test result show the value at $t = 0$ s, $n/n_0 = 0.07088$ and increase significantly at $t = 55$ s become 1.31444 and at $t = 150$ s become 167,21324. This is consistent with the statement that if $\rho > 0$ then the graph of the population of neutron flux equation versus a function of time tends to away from zero.

Keywords : Simulation, Reactivity, Characteristic Equation, Neutron Flux, Function of Time.

1. PENDAHULUAN

Pemanfaatan kemajuan teknologi di bidang komputer tak hanya berhenti sebagai suatu alat hitung sederhana. Komputer yang dioperasikan oleh manusia mampu memecahkan masalah perhitungan yang kompleks. Dengan memiliki pemahaman tentang bahasa komputer (bahasa pemrograman), maka seorang manusia bisa membangun suatu aplikasi yang mempermudah pekerjaannya.

Perhitungan yang kompleks dengan beberapa persoalan matematik yang tidak mungkin dikerjakan manusia secara cepat dapat dikerjakan melalui komputer dengan bantuan bahasa pemrograman dan pengetahuan seputar komputasi numerik. Kedua ilmu pengetahuan tersebut digabungkan untuk mewujudkan suatu bentuk maya dari program perhitungan yang kompleks.

Salah satu persoalan yang dapat dipecahkan oleh komputer adalah perhitungan populasi fluks neutron terhadap fungsi waktu. Dengan pemahaman yang baik tentang bahasa pemrograman dan urutan proses atau alur algoritma suatu program bekerja, maka dapat diciptakan suatu aplikasi yang mampu menyelesaikan persoalan tersebut dengan hasil yang secara langsung dapat tertampil dalam bentuk grafik.

2. DASAR TEORI

2.1. Kinetika Reaktor Nuklir

Kinetika reaktor nuklir adalah bagian dari fisika reaktor yang mempelajari perilaku gerakan reaksi inti berantai (gayut waktu) yang terjadi di dalam reaktor nuklir (Syarip, 2001). Tujuan dari pengendalian reaktor yakni mengatur dan mengendalikan proses reaksi pembelahan inti yang terjadi di dalam reaktor, sedemikian rupa sehingga

reaktor dapat dioperasikan mulai dari kondisi subskritik sampai pada tingkat daya, menaikkan dan mempertahankan daya reaktor pada tingkat daya tertentu, dan menurunkan, menghentikan atau mematikan operasi reaktor.

Kemampuan tingkat operasi pembangkitan daya dari suatu reaktor nuklir bergantung kepada seberapa besar kandungan massa bahan dapat belahnya, tampang lintang pembelahan makroskopik, dan besarnya fluks neutron. Dari ketiga faktor tersebut yang paling praktis dengan waktu tanggap cepat dan sensitivitas tinggi untuk pengendalian atau pengaturan daya reaktor adalah parameter fluks neutron. Mekanisme gerak dan perilaku perubahan fluks neutron terhadap fungsi waktu akan dibahas di dalam kinetika dan pengendalian reaktor nuklir secara detail. Dalam hal ini digunakanlah persamaan kinetika reaktor. Persamaan kinetika reaktor merupakan gambaran dari perilaku neutron (n, n_0) di dalam reaktor tiap satuan waktu akibat perubahan faktor perlipatan neutron di dalam reaktor (k) yang dipengaruhi oleh reaktivitas reaktor (ρ).

Hubungan antara faktor perlipatan neutron (k) dengan reaktivitas reaktor (ρ) ditunjukkan oleh persamaan berikut:

$$\rho = \frac{k-1}{k} = 1 - \frac{1}{k} \text{ atau } k = \frac{1}{1-\rho} \quad \dots (1)$$

dengan reaktivitas reaktor (ρ) diperoleh dari persamaan per-jam,

$$\rho = \frac{\ell}{\ell + T} + \frac{T}{\ell + T} \sum_{i=1}^6 \frac{\beta_i}{1 + \lambda_i T} \quad \dots (2)$$

Reaktivitas bisa berharga positif, bisa berharga negatif atau berharga nol. Reaktivitas negatif berarti $k < 1$ maka kondisi reaktor subkritis. Reaktivitas positif berarti $k > 1$ maka kondisi reaktor superkritis. Reaktivitas berharga nol berarti $k = 1$ maka kondisi reaktor kritis. Pengaturan reaktivitas di dalam pengendalian reaktor dilakukan dengan cara penyisipan dan panarikan batang kendali ke atau dari dalam teras reaktor.

Persamaan per-jam dapat juga diturunkan dengan menerapkan transformasi Laplace pada persamaan kinetika reaktor sebagai berikut:

$$sN - N_0 = k(\rho - \beta) \frac{N}{\ell} + \sum_{i=1}^6 \lambda_i C_i \quad \dots (3)$$

$$L \left[\frac{dC_i}{dt} \right] = L \left[k \beta_i \frac{n}{\ell} \right] - L [\lambda_i C_i]$$

menjadi,

$$sC_i - C_{i0} = k \beta_i \frac{N}{\ell} - \lambda_i C_i \quad \dots (4)$$

dimana S adalah variabel transformasi N_0 dan C_{i0} yang merupakan kerapatan neutron dan konsentrasi isotop penghasil neutron kasip pada kesetimbangan awal.

Pada keadaan kesetimbangan dimana $k = 1$, laju perubahan konsentrasi isotop-isotop penghasil neutron kasip tetap atau,

$$\frac{dC_i}{dt} = 0 \quad \dots (5)$$

maka diperoleh persamaan C_{i0} sebagai berikut:

$$C_{i0} = \beta_i \frac{N_0}{\lambda_i \ell} \quad \dots (6)$$

dari persamaan-persamaan di atas, diperoleh suatu harga N sebagai berikut:

$$N = \frac{\ell N_0 + \sum_{i=1}^6 \frac{\beta_i N_0}{\ell(s + \lambda_i)}}{\ell s - k(\rho - \beta) - k \sum_{i=1}^6 \frac{\beta_i \lambda_i}{s + \lambda_i}} \quad \dots (7)$$

$$N = N_0 \frac{\ell + \sum_{i=1}^6 \frac{\beta_i}{s + \lambda_i}}{\ell s + k \sum_{i=1}^6 \frac{s \beta_i}{s + \lambda_i} - (k - 1)} \quad \dots (8)$$

N merupakan hasil transformasi Laplace dari $n(t)$ atau dapat ditulis dengan $N = L[n(t)]$. Sehingga penyelesaian umum dari persamaan tersebut diperoleh,

$$\frac{n(t)}{n_0} = \sum_{j=1}^6 A_j e^{\omega_j t} \quad \dots (9)$$

dimana omega j (ω_j) merupakan akar-akar dari persamaan karakteristik yang diselesaikan dari persamaan sebagai berikut:

$$\ell s + k \sum_{i=1}^6 \frac{s \beta_i}{s + \lambda_i} - (k - 1) = 0 \quad \dots (10)$$

dimana S merupakan variabel transformasi N_0 dan C_{i0} yaitu kerapatan neutron dan konsentrasi isotop penghasil neutron kasip pada kesetimbangan awal. Dan koefisien A_j merupakan residu-residu yang besarnya,

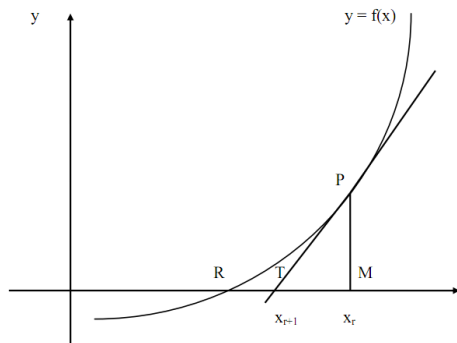
$$A_j = \frac{\ell + \sum_{i=1}^6 \frac{\beta_i \lambda_i}{s + \lambda_i}}{s + k \sum_{i=1}^6 \frac{\beta_i \lambda_i}{(s + \lambda_i)^2}} \bigg|_{s=\omega_j} \quad \dots (11)$$

Bila reaktivitas bernilai < 0 maka semua akar persamaan karakteristik bernilai negatif dan grafik populasi fluks neutron terhadap waktu cenderung mendekati nol. Bila reaktivitas bernilai > 0 maka akan ada satu akar persamaan karakteristik bernilai positif dan grafik populasi fluks neutron terhadap waktu cenderung menjauhi nol.

2.2. Metode Newton Raphson

Metode Newton Raphson merupakan metode yang umum dikenal untuk menyelesaikan fungsi $f(x) = 0$ (Supriyono, 2008). Metode ini digambarkan dalam grafik pada gambar 1. Gambar tersebut menunjukkan grafik fungsi $y=f(x)$ yang berpotongan dengan sumbu x pada titik R sebagai akarnya. Pendekatan langsung terhadap akarnya adalah x_r , yang memberikan titik P pada kurva tersebut. Dalam

kurva tersebut, ditambah suatu kurva tangensial terhadap kurva di titik P yang memotong sumbu di titik T . Bila jarak antara P dengan R kecil, kurva tidak akan menyimpang terlalu jauh dari garis lurus yang ada di dalam interval ini. Dengan demikian, T akan semakin dekat kepada R . Posisi T diambil sebagai pendekatan berikutnya terhadap akar X_{r+1} .



Gambar 1. Pendekatan Metode Newton Raphson

Sekarang tinggi PM adalah $f(x_r)$ dan $\tan(PM) = f'(x_r)$, maka secara trigonometri sederhana, terwujud suatu persamaan sebagai berikut.

$$x_{r+1} = x_r - \frac{f(x_r)}{f'(x_r)}$$

Rumusan ini merupakan rumus umum metode Newton Raphson. Dengan menemukan nilai pendekatan x_r , x_{r+1} dan seterusnya, dilakukan perhitungan hingga ditemukan nilai x yang memberikan nilai sama dengan nol pada fungsi $f(x)$ atau dengan kata lain, iterasi dilakukan sampai memberikan harga fungsi $f(x)$. Dalam perhitungan, nilai nol dapat ditoleransi dengan suatu nilai yang dianggap nol misalnya saja 1×10^{-15} .

Metode newton raphson merupakan salah satu metode dengan tingkat akurasi tinggi dalam menentukan akar persamaan polynomial derajat dua keatas. Selain itu, keunggulan metode ini adalah memiliki laju konvergensi kuadratik, sehingga metode ini lebih cepat untuk konvergen menuju akar pendekatan daripada metode lain yang memiliki laju konvergensi linear (Munir 2003).

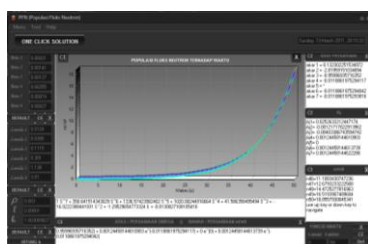
3. PROFIL PROGRAM

3.1. Nama Program

PFN (Populasi Fluks Neutron)

3.2. Tampilan Umum Program

Tampilan program PFN 1.1 secara umum dapat dilihat pada gambar 2 sebagai berikut.



Gambar 2. Tampilan umum PFN 1.1

3.3. Dimensi dan Ukuran File Program Utama

916 pixel × 620 pixel

3,251 MegaByte

4. METODE PENYELESAIAN PROGRAM

4.1. Penyusunan Persamaan Karakteristik

Persamaan karakteristik disusun dari persamaan (9) dengan penjabaran *sigma* pada suku kedua persamaan di atas sebagai berikut:

$$\sum_{i=1}^6 \frac{s\beta_i}{s+\lambda_i} = \frac{s\beta_1}{s+\lambda_1} + \frac{s\beta_2}{s+\lambda_2} + \dots + \frac{s\beta_6}{s+\lambda_6} \quad \dots(12)$$

Maka, dari penjabaran tersebut diperoleh persamaan sebagai berikut:

$$\begin{aligned} & s(s+\lambda_1)(s+\lambda_2)(s+\lambda_3)(s+\lambda_4)(s+\lambda_5)(s+\lambda_6) \\ & + \frac{k}{\ell} s\beta_1(s+\lambda_2)(s+\lambda_3)(s+\lambda_4)(s+\lambda_5)(s+\lambda_6) \\ & + \frac{k}{\ell} s\beta_2(s+\lambda_1)(s+\lambda_3)(s+\lambda_4)(s+\lambda_5)(s+\lambda_6) \\ & \dots \\ & + \frac{k}{\ell} s\beta_6(s+\lambda_1)(s+\lambda_2)(s+\lambda_3)(s+\lambda_4)(s+\lambda_5) \\ & - \frac{k-1}{\ell} (s+\lambda_1)(s+\lambda_2)(s+\lambda_3)(s+\lambda_4)(s+\lambda_5)(s+\lambda_6) = 0 \end{aligned} \quad \dots(13)$$

4.2 Pemodelan Matematik dalam Penyusunan Persamaan Karakteristik

4.2.1. Penggunaan Permutasi dalam Menentukan Nilai Variabel Berpangkat

Dalam menentukan nilai variabel berpangkat dari suatu persamaan berpangkat (misal penentuan nilai a , b dan c dalam persamaan $f(x) = ax^2 + bx + c$) dapat menggunakan operasi matematik yang disebut permutasi. Hasil dari operasi permutasi menunjukkan jumlah operasi penjumlahan dari komponen penyusun, bisa berupa $\lambda_{(1-6)}$ maupun operasi perkalian antar λ . Operasi permutasi dijabarkan sebagai berikut.

1). Permutasi persamaan berpangkat 6.

- ${}^6P = \frac{6!}{0!} = 1$ permutasi untuk harga variabel berpangkat 6. Dalam program dideklarasikan sebagai variabel **ka**.
- ${}^6P = \frac{6!}{1!1!} = 6$ permutasi untuk harga variabel berpangkat 5. Dalam program dideklarasikan sebagai variabel **la**.
- ${}^6P = \frac{6!}{2!2!} = 15$ permutasi untuk harga variabel berpangkat 4. Dalam program dideklarasikan sebagai variabel **ma**.
- ${}^6P = \frac{6!}{3!1!} = 20$ permutasi untuk harga variabel berpangkat 3. Dalam program dideklarasikan sebagai variabel **na**.
- ${}^6P = \frac{6!}{2!4!} = 15$ permutasi untuk harga variabel berpangkat 2. Dalam program dideklarasikan sebagai variabel **oa**.

- ${}^6_5P = \frac{6!}{1! \cdot 5!} = 6$ permutasi untuk harga variabel berpangkat 1. Dalam program dideklarasikan sebagai variabel **pa**.
- ${}^6_6P = \frac{6!}{6!} = 1$ permutasi untuk harga variabel berpangkat 0 atau disebut juga bilangan tak berpangkat. Dalam program dideklarasikan sebagai variabel **qa**.

Untuk tujuh variabel di atas, masing-masing berisi data λ .

2). Permutasi persamaan berpangkat 5.

- ${}^5_0P = \frac{5!}{5!} = 1$ permutasi untuk harga variabel berpangkat 5. Dalam program dideklarasikan sebagai variabel **ba**.
- ${}^5_1P = \frac{5!}{4! \cdot 1!} = 5$ permutasi untuk harga variabel berpangkat 4. Dalam program dideklarasikan sebagai variabel **bb**.
- ${}^5_2P = \frac{5!}{3! \cdot 2!} = 10$ permutasi untuk harga variabel berpangkat 3. Dalam program dideklarasikan sebagai variabel **bc**.
- ${}^5_3P = \frac{5!}{2! \cdot 3!} = 10$ permutasi untuk harga variabel berpangkat 2. Dalam program dideklarasikan sebagai variabel **bd**.
- ${}^5_4P = \frac{5!}{1! \cdot 4!} = 5$ permutasi untuk harga variabel berpangkat 1. Dalam program dideklarasikan sebagai variabel **be**.
- ${}^5_5P = \frac{5!}{5!} = 1$ permutasi untuk harga variabel berpangkat 0 atau disebut juga bilangan tak

berpangkat. Dalam program dideklarasikan sebagai variabel **bf**.

Untuk enam variabel tersebut, masing-masing berisi β dan λ dengan variabel B1 sampai B6 merupakan nilai masukan dari β ; l1 sampai l6 merupakan nilai masukan dari λ ; **kx** merupakan konstanta kekritisitas; **lx** merupakan umur neutron; **j** merupakan suku terakhir dari 3 suku pada persamaan 14 dan **z** merupakan pengali *sigma* pada suku kedua persamaan.

4.2.2. Penyusunan Persamaan Karakteristik

Perkalian bertingkat di setiap suku disederhanakan dengan variabel penyusun. Penyusunan persamaan dibantu dengan metode permutasi dengan pengelompokan berdasar tiap suku dari persamaan 10 yang sebelumnya telah dijabarkan pada persamaan 13 dapat dilihat pada gambar 3.

Dari proses pada gambar 3 diperoleh penyederhanaan persamaan 10 menjadi persamaan pangkat 7 menjadi fungsi,

$$f(S) = aS^7 + bS^6 + cS^5 + dS^4 + eS^3 + fS^2 + gS + h = 0$$

dengan :

a := la;
b := lb + (z*ba) - (j*la);
c := lc + (z*bb) - (j*lb);
d := ld + (z*bc) - (j*lc);
e := le + (z*bd) - (j*ld);
f := lf + (z*be) - (j*le);
g := lg + (z*bf) - (j*lf);
h := -1*(j*lg);

Suku 1 :	la(S ⁷) +	lb(S ⁶) +	lc(S ⁵) +	ld(S ⁴) +	le(S ³) +	lf(S ²) +	lg(S) +	
Suku 2 :		z*ba(S ⁶) +	z*bb(S ⁵) +	z*bc(S ⁴) +	z*bd(S ³) +	z*be(S ²) +	z*bf(S) +	
Suku 3 :		j*la(S ⁶) +	j*lb(S ⁵) +	j*lc(S ⁴) +	j*ld(S ³) +	j*le(S ²) +	j*lf(S) +	j*lg
Variabel	a	b	c	d	e	f	g	h

Gambar 3. Proses Pembentukan Persamaan Karakteristik

4.2. Metode Penyelesaian Akar Persamaan Karakteristik dengan Menggunakan Newton Raphson Bertingkat

Metode Newton Raphson hanya akan memberikan satu akar di setiap satu nilai iterasi awal yang ditentukan. Dengan memberikan nilai iterasi awal yang berbeda, memungkinkan untuk mendapatkan akar yang berbeda pula, mendekati nilai yang diberikan. Jika nilai awal iterasi diperoleh dari bilangan yang terletak di antara dua akar yang telah ditemukan (X1 dan X2) dari dua nilai awal iterasi yang ditentukan (N1 dan N2), dan mengulangnya hingga empat tingkat, daerah antara nilai batas tersebut akan dicacah lagi untuk mendapatkan pendekatan akar persamaan

karakteristik yang lebih akurat. Penjabarannya dapat dilihat pada gambar 4.

Dari gambar 4, metode Newton Raphson diulang hingga tujuh belas kali sampai pada level keempat. Bukan tidak mungkin dari X1 sampai X17 tidak ada bilangan yang sama setelah mengalami pembulatan hingga ketelitian tertentu (1×10^{-12} seperti pada program PFN 1.1) dan dimungkinkan terdapat akar-akar imajiner yang tidak bisa dihitung dalam perhitungan program (tidak masuk dalam range bilangan yang digunakan dalam memproses masukan ke dalam suatu perhitungan : *Real*) sehingga ditambah perintah dalam program untuk memunculkan akar-akar bukan bilangan imajiner dan hanya muncul akar-akar yang tidak sama.

lv.I		X1															X2
lv.II				X5		X3		X4									
lv.III	N1		X9		X8		X7		X6								N2
lv.IV			X17	X16	X15	X14	X13	X12	X11	X10							

Gambar 4. Proses Pengulangan Newton Raphson Empat Tingkat

4.3. Penyelesaian Populasi Fluks Neutron terhadap Fungsi Waktu

Akar-akar persamaan karakteristik yang diperoleh merupakan nilai dari w_j dengan j merupakan indeks dari akar persamaan (1-7). Banyaknya residu (A_j) mengikuti banyaknya akar persamaan, sedangkan nilai residu ditelusuri dalam program dengan mengikuti persamaan 11. Kedua variabel tersebut merupakan komponen utama penyusun persamaan populasi fluks neutron terhadap fungsi waktu. Dari kedua variabel tersebut ditambah dengan waktu yang ditentukan oleh pengguna, nilai dan grafik populasi fluks neutron terhadap fungsi waktu diselesaikan melalui kode program sebagai berikut:

for u := ta to tb **do**

begin

t[u]:=u;

nt[u]:=(A1*(Exp((w1*t[u])))+A2*(Exp((w2*t[u])))+A3*(Exp((w3*t[u])))+A4*(Exp((w4*t[u])))+A5*(Exp((w5*t[u])))+A6*(Exp((w6*t[u])))+A7*(Exp((w7*t[u]))));

fNtpn0.memo1.lines.add('n'+inttostr(u)+'='+

floattostr(nt[u]));

fGrafik.chart1.series[0].addxy(t[u], nt[u],",cled);

end;

dengan t adalah fungsi waktu; ta adalah waktu awal dan tb adalah waktu akhir; nt adalah populasi fluks neutron terhadap fungsi waktu (n/n_0).

5. HASIL DAN PEMBAHASAN

Kesesuaian program terhadap dasar teori (proses pengujian program) perlu dilakukan untuk membuktikan kebenaran hasil perhitungan dari program simulasi. Dalam pengujian program, dimasukkan data reaktivitas positif, nol dan negatif, serta beberapa variabel pendukung seperti yang

ditunjukkan pada tabel 1. Sedangkan untuk data β dan λ dari isotope U^{235} , tertera dalam tabel 2.

Tabel 1. Data masukan untuk reaktivitas positif, nol dan negatif

Variabel	Sisipan Reaktivitas		
	Positif	Nol	Negatif
ρ	0,002	0	-0,002
ℓ	0,001 s	0,001 s	0,001 s
ta	0 s	0 s	0 s
tb	150 s	150 s	150 s

Tabel 2. Data Beta dan Lamda

No	Beta(β)	Lamda(λ)
1	0.00021	0.0124667
2	0.00141	0.0305351
3	0.00127	0.1114384
4	0.00255	0.3013683
5	0.00074	1.1363068
6	0.00027	3.0136833

Setelah dikalkulasi melalui program, diperoleh persamaan karakteristik sebagai berikut:

1). Untuk Sisipan Reaktivitas Positif

$$S^7 + 9.06232 S^6 + 23.27738 S^5 + 17.32533 S^4 + 2.35678 S^3 - 0.00944 S^2 - 0.00737 S - 8.71307 \dots(14)$$

2). Untuk Sisipan Reaktivitas Nol

$$S^7 + 649.6034S^6 + 2713.60062S^5 + 2626.83207S^4 + 574.172021S^3 - 33.13793S^2 - 0.364247S \dots(15)$$

3). Untuk Sisipan Reaktivitas Negatif

$$S^7 + 748.85914S^6 + 3170.77521S^5 + 3160.27509S^4 + 750.99117S^3 - 49.45918S^2 - 0.915217S - 0.00434 \dots(16)$$

Tabel 3. Akar-akar Persamaan (w_j) dan Residu (A_j)

No.	Reaktivitas Positif		Reaktivitas Nol		Reaktivitas Negatif	
	w_j	A_j	w_j	A_j	w_j	A_j
1	0.05101	0.07945	0	0.08894	-0.00933	0.02308
2	-5.22611	-4.49641	-6.94457	-7.28012	-8.79480	-7.25036
3	-2.70329	-0.00100	-2.82801	-0.00164	-2.88296	-0.00231
4	-0.13172	-0.00322	-0.19057	-0.06903	-0.22748	-0.15356
5	-0.96157	-0.00514	<i>imaginer</i>	-	<i>imaginer</i>	-
6	-0.01216	0.00042	-0.19057	-0.06903	-0.06320	0.12956
7	-0.01216	0.00042	0	0.08894	-0.02466	0.01060

Akar-akar persamaan dan residu tertera pada tabel 3. Populasi fluks neutron terhadap fungsi waktu ditunjukkan dalam tabel 4. Dan grafik populasi fluks neutron terhadap fungsi waktu secara berturut-turut untuk sisipan reaktivitas positif, nol dan negatif ditunjukkan dalam gambar 5.

Dalam penentuan akar-akar persamaan karakteristik (persamaan 14, 15 dan 16) digunakan metode newton raphson. Metode ini merupakan salah satu metode yang memiliki akurasi tinggi untuk memperoleh akar persamaan polinomial derajat dua keatas. Metode ini memiliki tingkat

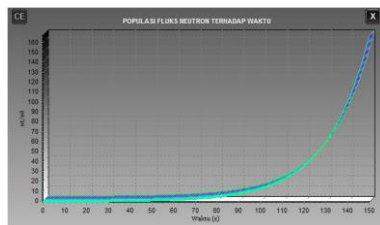
ketelitian hingga 1×10^{-13} (Mengacu pada program yang dibuat) dalam memisahkan akar-akar persamaan yang berbeda. Selain itu, metode ini memiliki laju konvergen yang cepat menuju pendekatan akar persamaan.

Dengan masukan data reaktivitas positif data reaktivitas nol, salah satu w_j bernilai negatif dan populasi fluks neutron naik seiring bertambahnya waktu sehingga grafik yang terbentuk cenderung menjauhi nol. Untuk masukan data reaktivitas nol, hasil dari pengujian menunjukkan bahwa terdapat w_j bernilai nol dan populasi fluks neutron bernilai

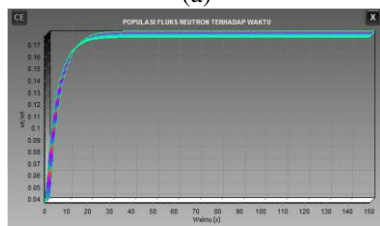
konstan (kisaran 0.17787) dari $t = 55$ hingga $t = 150$ dan grafik populasi fluks neutron cenderung mendatar. Sedangkan, dengan masukan reaktivitas negatif, semua w_j bernilai negatif dan populasi fluks neutron setelah mencapai jumlah tertentu akan turun seiring bertambahnya waktu sehingga grafik yang terbentuk cenderung mendekati nol. Hal ini sesuai dengan pernyataan bahwa bila $\rho > 0$ maka akan ada satu akar persamaan karakteristik bernilai positif dan grafik populasi fluks neutron terhadap waktu cenderung menjauhi nol. Bila $\rho < 0$ maka semua akar persamaan karakteristik bernilai negatif dan grafik populasi fluks neutron terhadap waktu cenderung mendekati nol. Hasil ini menunjukkan program dapat berjalan sesuai dengan sisipan reaktivitas yang diberikan.

Tabel 4. Populasi Fluks Neutron terhadap Fungsi Waktu

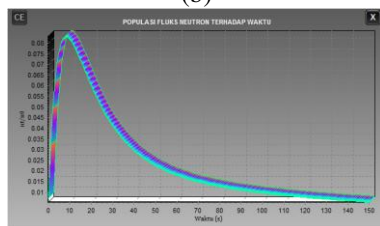
t (s)	n_t/n_0 ($\rho > 0$)	n_t/n_0 ($\rho = 0$)	n_t/n_0 ($\rho < 0$)
0	0.07088	0.03810	0.00729
1	0.07958	0.06367	0.03238
2	0.08558	0.08356	0.04948
..
8	0.11913	0.14781	0.08338
..
55	1.31444	0.17787	0.02055
..
150	167.21324	0.17787	0.00596



(a)



(b)



(c)

Gambar 5. Grafik Populasi Fluks Neutron terhadap Fungsi Waktu

Program ini memiliki kelemahan dalam penyajian hasil n_t/n_0 , dengan tipe data *Real* dengan

range $5.0 \times 10^{-324} \dots 1.7 \times 10^{308}$, maka hasil perhitungan tak bisa melebihi range tersebut. Selain itu, jika terlalu kecil dalam menentukan nilai reaktivitas, akan sulit untuk mengamati grafik n_t/n_0 terhadap t , karena penurunan populasi fluks neutron akan berjalan sangat lambat seiring bertambahnya waktu.

6. KESIMPULAN

- 1). Program simulasi perhitungan populasi fluks neutron di dalam teras reaktor dapat berjalan sesuai dengan sisipan reaktivitas yang diberikan.
- 2). Permutasi dapat digunakan untuk menentukan jumlah operasi penjumlahan dalam proses penyusunan persamaan karakteristik yang merupakan persamaan polynomial derajat tujuh.
- 3). Metode Newton Raphson membuat program menentukan akar persamaan karakteristik secara cepat dengan ketelitian hingga 1×10^{-13} .
- 4). Dengan metode Newton Raphson bertingkat, maka diperoleh pendekatan akar persamaan karakteristik yang lebih akurat (pendekatan hingga 17 akar persamaan berbeda).

DAFTAR PUSTAKA

- Munir, Rinaldi. 2003. *Metode Numerik*. Informatika : Bandung.
- Supriyono. 2008. *Pemrograman Komputer dan Komputasi Numerik*. Sekolah Tinggi Teknologi Nuklir-Badan Tenaga Nuklir Nasional : Yogyakarta.
- Syarip. 2001. *Pengenalan Kinetika dan Pengendalian Reaktor Nuklir*. Pusat Penelitian dan Pengembangan Teknologi Maju -Badan Tenaga Nuklir : Yogyakarta.